ลิขิต ดลชัย

การใช้ หน่วยความจำ ในโลตัส 1-2-3

หลักการใช้หน่วยความจำปกติของโลตัส 1-2-3 อย่างมีประสิทธิภาพ และวิธีปฏิบัติเมื่อหน่วยความจำเหลือน้อย

เมื่อเราเรียกโปรแกรมโลตัส 1-2-3 มาใช้งาน ด้วยการโหลดลงสู่หน่วยความจำ โปรแกรมโลตัส 1-2-3 จะใช้พื้นที่หน่วยความจำส่วนหนึ่ง และเหลือพื้นที่ว่างอีกส่วนสำหรับการสร้างและแก้ไขชิ้นงานเวอร์กชีต ถ้ามีพื้นที่ว่างมาก เราก็สามารถสร้างเวอร์กชีตขนาดใหญ่ ๆ ได้ แต่ถ้ามีพื้นที่ว่างน้อยขณะที่กำลังสร้างเวอร์กชีตอยู่ ก็อาจมีสถานภาพเตือนว่าหน่วยความจำเหลืออีกเพียงนิดเดียว หากยังทำการเพิ่มข้อมูลเข้าในเวอร์กชีตอีกจนล้นหน่วยความจำที่มีอยู่ โปรแกรมจะขึ้นข้อความบอกความผิดพลาดว่าหน่วยความจำเต็ม

พื้นที่ว่างสำหรับใช้สร้างและแก้ไขเวอร์กชีตของหน่วยความจำปกติ หรือหน่วยความจำตั้งแต่ตำแหน่ง 0 ไปถึงตำแหน่ง 640 กิโลไบต์ จะควบคุมให้เหลือมากหรือน้อยขึ้นอยู่กับการสั่งหรือใช้งานของผู้ใช้เองด้วย คือจะขึ้นกับขนาดของโปรแกรมดำเนินงานดอสว่าใช้ดอสเวอร์ชันอะไร ถ้าเป็นดอสเวอร์ชันใหม่ ๆ ขนาดของโปรแกรมดอสจะใหญ่โตกินเนื้อที่มากกว่าดอสเวอร์ชันเก่า ๆ แต่ถ้าเป็นเอ็มเอสดอสเวอร์ชัน 5 หรือดีอาร์ดอสเวอร์ชัน 5 หรือ 6 เราอาจย้ายบางส่วนของดอสไปใช้พื้นที่หน่วยความจำยืดขยายสำหรับเครื่องตั้งแต่รุ่นเอที 286 ขึ้นไป ทำให้พื้นที่ว่างของหน่วยความจำปกติมีมากขึ้น ถัดไปจะขึ้นกับว่าเรามีการเรียกใช้โปรแกรมประเภทฝังในหน่วยความจำอื่น ๆ ก่อนการเรียกใช้โปรแกรมโลตัส 1-2-3 อีกหรือไม่เพียงไร โปรแกรมประเภทฝังนี้จะจองพื้นที่อยู่ด้วยเสมอจนกว่าจะยกเลิกหรือปิดเครื่องเลิกงาน

การเรียกโปรแกรมโลตัส 1-2-3 มาโหลดลงสู่หน่วยความจำ จะเปลืองเนื้อที่หน่วยความจำมากหรือน้อย จะขึ้นกับเราเลือกใช้ไดรเวอร์ขนาดใหญ่หรือเล็ก ขณะที่เราทำการติดตั้งหรือปรับระบบของ 1-2-3 เช่น เลือกให้มีลักษณะ UNDO ของการยกเลิกงานใหม่และเรียกงานเดิมกลับคืนมาไว้ด้วยหรือไม่ มีการเลือกใช้งานโหมด WYSIWYG ด้วยหรือไม่ การเลือกมีลักษณะ UNDO และใช้งานโหมด WYSIWYG จะสิ้นเปลืองหน่วยความจำมากจนมีที่ว่างเหลือสำหรับเวอร์กชีตน้อยลง

ในการสร้างและแก้ไขเวอร์กชีต ลักษณะหรือโปรแกรมประเภทของข้อมูลที่บรรจุในเซลของเวอร์กชีต ไม่ว่า

จะเป็นข้อมูลตัวอักษร ตัวเลข หรือสูตรคำนวณ จะใช้เนื้อที่หน่วยความจำไม่เท่ากัน การจัดองค์ประกอบของการบรรจุข้อมูลลงสู่เซลในบริเวณต่าง ๆ ก็มีผลต่อการใช้เนื้อที่หน่วยความจำด้วย

มุมสเปรดชีตฉบับนี้จะได้กล่าวถึงหลักการใช้หน่วยความจำของโลตัส 1-2-3 เพื่อช่วยให้เรามีการใช้หน่วยความจำสำหรับโลตัส 1-2-3 อย่างมีประสิทธิภาพ ขณะเดียวกันจะบอกวิธีควรปฏิบัติกรณีที่พบปัญหาว่าหน่วยความจำเหลือน้อยจนไม่พอกับการใช้งานต่อไป โดยในตอนต้นจะกล่าวถึงประเภทของหน่วยความจำ และการใช้หน่วยความจำนั้น ๆ ของโลตัส 1-2-3

ประเภทของหน่วยความจำ

เมื่อเครื่องไมโครคอมพิวเตอร์แบบ 16 บิต ตระกูลไอบีเอ็มพีซี ที่มีซีพียูเป็นไมโครโปรเซสเซอร์ 8088 ออกมาใหม่ ๆ โปรแกรมเอ็มเอสดอสที่ใช้ควบคุมงานต่าง ๆ บนเครื่องนี้ จะแบ่งหน่วยความจำของเครื่องเป็นสองส่วน ส่วนแรกเรียก หน่วยความจำปกติ (Conventional Memory Area) ตั้งแต่ตำแหน่ง 0 เรื่อยไปจนถึง 640 กิโลไบต์ ส่วนที่สองเรียก หน่วยความจำตอนบน (Upper Memory Area) ตั้งแต่ตำแหน่ง 640 กิโลไบต์ไปจนถึงตำแหน่ง 1 เมกะไบต์ (รวม 384 กิโลไบต์) สำหรับเป็นหน่วยความจำเพื่อติดต่อแสดงผลที่จอภาพ และหน่วยความจำเก็บโปรแกรมติดต่ออุปกรณ์อินพุตเอาต์พุต โปรแกรมตรวจสอบฮาร์ดแวร์ และโปรแกรมตัวแปลภาษาเบสิก ซึ่งโปรแกรมโลตัส 1-2-3 และการสร้างแก้ไขเวอร์กชีตจะใช้งานกับหน่วยความจำปกติตอนแรกเท่านั้น

เนื่องจากการสร้างแก้ไขเวอร์กชีตเป็นการติดต่อและเรียกใช้หน่วยความจำปกติตลอด ถ้าเวอร์กชีตใหญ่มากขึ้น ๆ หน่วยความจำปกติจึงไม่พอกับการใช้งาน เช่นเดียวกับโปรแกรมประยุกต์ใช้งานอื่น ๆ เช่น โปรแกรมดีเบส ก็ประสบปัญหาที่มีหน่วยความจำปกติไม่พอ ดังนั้นบริษัท โลตัสดีเวลลอปเม้นท์ บริษัท อินเทล และบริษัท ไมโครซอฟต์ จึงได้ร่วมกันออกข้อกำหนดของหน่วยความจำเพิ่มขยาย (Expanded Memory Specification หรือ EMS) เรียกย่อ ๆ ว่า LIM ด้วยการติดตั้งซอฟต์แวร์ที่เป็นไดรเวอร์เรียก expanded memory manager และแผงหน่วยความจำเพิ่มขยายสำหรับเสียบลงในช่องสล้อตของเครื่องพีซี การอ่านเขียนหรือเรียกติดต่อกับหน่วยความจำที่เพิ่มเข้ามานี้จะเข้าออกผ่านตำแหน่งหน่วยความจำตอนบนที่ว่างไม่ได้นำไปใช้งาน ซึ่งการติดต่อนี้จะสลับเข้าออกทางช่องหน้าต่างเป็นหน้า ๆ หน้าละ 16 กิโลไบต์ ข้อกำหนดที่ใช้อยู่เดิมคือ เวอร์ชัน 3.2 แต่ปัจจุบันในเวอร์ชัน 4.0 แล้ว โปรแกรมโลตัส 1-2-3 มีการใช้หน่วยความจำเพิ่มขยายนี้มาใช้งานสร้างเวอร์กชีตที่ใหญ่ขึ้นกว่าเดิมที่ใช้ในหน่วยความจำปกติอย่างเดียว

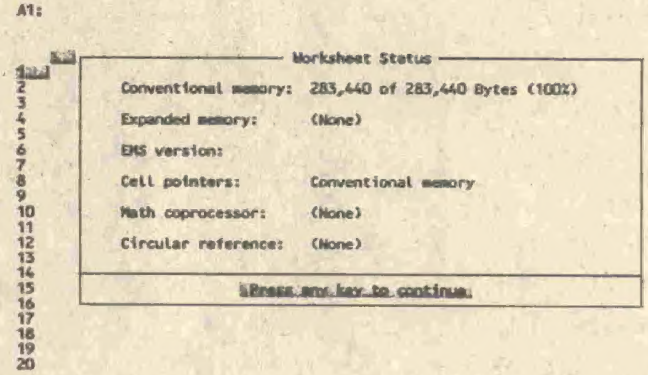
เครื่องไมโครคอมพิวเตอร์รุ่นพีซีได้มีการพัฒนาเจริญขึ้น ด้วยการใช้ไมโครโปรเซสเซอร์สูงขึ้น เป็นเครื่องรุ่นเอทีใช้ไมโครโปรเซสเซอร์ 80286 เครื่องใช้ไมโครโปรเซสเซอร์ 80386, 80386SX, 80486 และในปัจจุบันเริ่มมี 80486SX เครื่องต่าง ๆ เหล่านี้จะมีหน่วยความจำสูงกว่า 1 เมกะไบต์ หน่วยความจำที่สูงจาก 1 เมกะไบต์ออกไปจะเรียกว่าหน่วยความจำยืดขยาย (Extended Memory) โปรแกรมโลตัส 1-2-3 ยังไม่มีการนำหน่วยความจำยืดขยายนี้มาประยุกต์ใช้งานอย่างจริงจัง จะมีก็แต่เพียงใช้โปรแกรมยูทิลิตี้ เช่น เทอร์โบ EMS หรือ QEMM นำหน่วยความจำ

ยึดขยายมาจำลองใช้งานเป็นหน่วยความจำเพิ่มขยาย เพื่อใช้งานกับโลตัส 1-2-3 นอกจากนี้โปรแกรมยูทิลิตี้เหล่านี้ยังสามารถใช้พื้นที่ดิสค์มาจำลองเป็นหน่วยความจำเพิ่มขยายได้อีกด้วย แต่มีข้อเสียที่ทำงานค่อนข้างช้า

เอ็มเอสดอส เวอร์ชัน 5 มีไดรเวอร์ที่พร้อมให้ติดตั้ง นำโปรแกรมดอสบางส่วนไปโหลดไว้ที่หน่วยความจำยืดขยาย โดยเฉพาะบริเวณที่เป็น 64 กิโลไบต์แรก นับจากตำแหน่งที่ 1 เมกะไบต์เป็นต้นไป ซึ่งจะเรียกว่า หน่วยความจำสูง (High Memory Area) ทำให้พื้นที่หน่วยความจำปกติเหลือว่างมากขึ้น ขณะเดียวกันดอส 5 จะมีไดรเวอร์ให้แปลงหน่วยความจำยืดขยาย มาทำงานเป็นหน่วยความจำเพิ่มขยายได้อีกด้วย (รายละเอียดในเรื่องนี้วารสารไมโครฯ เคยนำเสนอไว้บ้างแล้ว ขอให้ค้นอ่านในคอลัมน์มุมโอเอสฉบับก่อน ๆ)

การเรียกใช้หน่วยความจำเพิ่มขยาย

เมื่อมีการติดตั้งหน่วยความจำเพิ่มขยายเข้ากับระบบ โปรแกรมโลตัส 1-2-3 จะรับรู้และแสดงสถานภาพของหน่วยความจำเพิ่มขยาย ด้วยการใช้คำสั่ง /Worksheet Status ดังรูปที่ 1 ซึ่งจะมีข้อมูลบอกหน่วยความจำปกติเหลืออยู่เท่าไร มีหน่วยความจำเพิ่มขยายหรือไม่ ถ้ามีจะเป็น EMS เวอร์ชันอะไรอาจเป็น LIM 3.2 หรือ LIM 4.0 ข้อมูลตำแหน่งตัวชี้เซลมีการจัดเก็บในหน่วยความจำประเภทไหน อาจเป็นหน่วยความจำปกติ หรือหน่วยความจำเพิ่มขยาย มีการใช้ตัวร่วมประมวลผลคณิตศาสตร์หรือไม่ และสุดท้ายมีการอ้างชี้เป็นวนรอบตัวเองหรือไม่



รูปที่ 1

ตามปกติแล้ว ถ้ามีการติดตั้งหน่วยความจำเพิ่มขยาย โปรแกรมโลตัส 1-2-3 จะเก็บข้อมูลในเซลไว้ที่หน่วยความจำเพิ่มขยาย และเก็บตำแหน่งตัวชี้เซลไว้ที่หน่วยความจำปกติ ซึ่งวิธีการจัดเก็บนี้เป็นแบบมาตรฐานที่ให้สมรรถภาพสูง และเข้าได้กับโปรแกรมรีลีสต่าง ๆ กับโปรแกรมต่อพ่วงประเภท Add-In ต่าง ๆ แต่ใน 1-2-3 รีลีส 2.3 ใหม่นี้ เอื้ออำนวยกับการใช้งานหน่วยความจำเพิ่มขยาย เวอร์ชัน LIM 4.0 มากขึ้น คือ ให้เรากำหนดใช้หน่วยความจำเพิ่มขยายเก็บทั้งข้อมูลในเซลและตำแหน่งตัวชี้เซล ด้วยการใช้คำสั่ง /Worksheet Global Default Other Expanded Memory Enhanced การปรับตั้งนี้จะทำให้หน่วยความจำปกติว่างมากขึ้น สร้างเวอร์กชีตขนาดใหญ่ ๆ ได้ แต่จะมีข้อเสียที่เวอร์กชีตใหญ่ ๆ นี้ใช้กับโลตัส 1-2-3 รีลีสเก่าไม่ได้ และใช้กับโปรแกรมต่อพ่วงบางตัวไม่ได้ ที่มองตัวชี้เซลในหน่วยความจำปกติเพียงที่เดียว ค่าดีฟอลต์ของการตั้งวิธีใช้หน่วยความจำเพิ่มขยายของโลตัส 1-2-3 รีลีส 2.3 แบบมาตรฐานปกติจะเป็น /Worksheet Global Default Other Expanded Memory Standard

ในโลตัส 1-2-3 รีลีส 2.3 เราสามารถใช้หน่วยความจำเพิ่มขยายสำหรับเก็บข้อมูลเวอร์กชีตได้สูงถึง 4 เมกะไบต์ เช่นเดียวกับโลตัส 1-2-3 รีลีสเก่า ๆ แต่ถ้าเราใช้หน่วยความจำเพิ่มขยายตามข้อกำหนด LIM 4.0 ด้วย เราสามารถใช้หน่วยความจำเพิ่มขยายสำหรับเก็บตำแหน่งตัวชี้เซลได้เพิ่มสูงถึง 8 เมกะไบต์ ดังนั้นขนาดของเวอร์กชีตอาจขยายได้สูงสุดถึง 12 เมกะไบต์ ซึ่งถือว่าใหญ่มาก ๆ

การเก็บข้อมูลในหน่วยความจำ

ถ้าในระบบเครื่องของเรามีเพียงหน่วยความจำปกติ โลตัส 1-2-3 จะเก็บงานทุกอย่างในหน่วยความจำปกตินี้ แต่ถ้ามีการติดตั้งหน่วยความจำเพิ่มขยาย ข้อมูลบางส่วนก็จะแบ่งไปเก็บในหน่วยความจำเพิ่มขยายด้วย

ข้อมูลที่เก็บในหน่วยความจำปกติได้แก่

- ตัวโปรแกรม โลตัส 1-2-3 เอง
- ชื่อกลุ่มเซล และชื่อกราฟ ที่มีการกำหนดชื่อ
- ผลเนื่องจากการใช้สูตรแสดง

ตัวอักษร

- ชุดไดรเวอร์
- โปรแกรมต่อพ่วงประเภท Add-In ต่าง ๆ
- โปรแกรมฝังในหน่วยความจำ เช่น โปรแกรม BPRINT เพื่อการพิมพ์เวอร์กชีตที่หลังฉาก
- โปรแกรมเครือข่าย
- ฟอนต์แบบ WYSIWYG (ที่ขนาดโตกว่า 16 กิโลไบต์)
- ตำแหน่งตัวชี้เซล (เมื่อใช้หน่วยความจำเพิ่มขยายแบบมาตรฐานปกติ)

ข้อมูลที่เก็บในหน่วยความจำเพิ่มขยาย ได้แก่

- ค่าตัวอักษร
- สูตรคำนวณ
- ค่าตัวเลขที่มีจุดทศนิยมจาก - 32767 ถึง +32767
- บัฟเฟอร์ของการทำ UNDO
- ฟอนต์แบบ WYSIWYG (ที่มีขนาดเล็กกว่า 16 กิโลไบต์)
- ตำแหน่งตัวชี้เซล (เมื่อมีการใช้หน่วยความจำเพิ่มขยายแบบเพิ่มเติมและตามข้อกำหนด LIM 4.0)

เนื่องจากหน่วยความจำเพิ่มขยายอาจมีขนาดจำกัด ดังนั้นโอกาสที่หน่วยความจำเพิ่มขยายจะไม่พอเพียงสำหรับการเก็บข้อมูลจึงเป็นไม่ได้มาก ตัวอย่างเช่น ถ้าขนาดของบัฟเฟอร์ UNDO การทำงานเรียกงานเดิมกลับคืนมา ไม่สามารถบรรจุในหน่วยความจำเพิ่มขยาย โลตัส 1-2-3 จะบรรจุบัฟเฟอร์ในหน่วยความจำเพิ่มขยายจนเต็มเสียก่อน แล้วจึงนำที่เหลือไปเก็บในหน่วยความจำปกติ หรือในกรณีการเก็บค่าตัวอักษร สูตรคำนวณหรือค่าตัวเลขทศนิยม ในหน่วยความจำเพิ่มขยายจนเต็ม 1-2-3 จะทำการเก็บค่าต่อไปในหน่วยความจำปกติ

ตำแหน่งตัวชี้เซล หรือเซลพอยน์เตอร์ ที่กล่าวถึงข้างต้น จะเป็นตัวชี้ที่บอกตำแหน่งข้อมูลที่เก็บในหน่วยความจำ ที่อาจเป็นหน่วยความจำปกติหรือหน่วยความจำเพิ่มขยาย เซลหนึ่ง ๆ ในเวอร์กชีตจะใช้พื้นที่หน่วยความจำ 4 ไบต์ เพื่อเก็บตำแหน่งตัวชี้เซล ซึ่งถ้าค่าของเซลเป็นตัวเลขจำนวนเต็มระหว่าง -32767 ถึง +32767 โลตัส 1-2-3 จะเก็บค่าโดยตรงกับตำแหน่งตัวชี้เซล ไม่จำเป็นต้องใช้พื้นที่หน่วยความจำเพิ่มอีก

การจัดหน่วยความจำของ 1-2-3

ในการป้อนข้อมูลลงสู่เซลของเวอร์กชีต โลตัส 1-2-3 รีลีส 2.3 จะใช้พื้นที่หน่วยความจำเป็นบล็อก บล็อกละสี่เซลตามแนวคอลัมน์ ตัวอย่างเช่น เซล A1, A2, A3 และ A4 ถือเป็นข้อมูลบล็อกเดียวกัน เซล A5, A6, A7 และ A8 จัดเป็นข้อมูลอีกบล็อกหนึ่ง ในแต่ละเซลจะใช้เนื้อที่ 4 ไบต์ เพื่อเก็บตำแหน่งตัวชี้เซลและจำนวนไบต์เพิ่มเติมอีก ตามชนิดของข้อมูลที่ป้อนลงสู่เซล ตัวอย่างเช่น ถ้าเราป้อนค่า 28 ที่เซล A1 และค่า 64 ที่เซล A5 หน่วยความจำที่ใช้ไปจะเป็น 32 ไบต์ บล็อกใดที่มีการป้อนค่าลงไปบล็อกนั้นจะเกิดแอคตีฟทันที ตัวอย่างข้างต้นเป็นการป้อนค่าตัวเลขจำนวนเต็มระหว่าง -32767 และ +32767 ดังนั้นจึงไม่มีการใช้หน่วยความจำเพิ่ม แต่ถ้าป้อนข้อมูลตัวอักษร ข้อมูลตัวเลขที่มีจุดทศนิยม หรือสูตรคำนวณอื่น ๆ ก็จะต้องใช้หน่วยความจำเพิ่มมากขึ้นตามชนิดของข้อมูล ตารางในรูปที่ 2 จะเป็นหน่วยความจำที่เพิ่มมากขึ้น จากการกันหรือสำรองหน่วยความจำของตำแหน่งตัวชี้เซลปกติ (16 ไบต์ต่อบล็อก) แล้ว

รูปที่ 2 การใช้หน่วยความจำเพิ่มมากขึ้น

การป้อนข้อมูลในเวอร์กชีต	ขนาดหน่วยความจำเพิ่มขึ้นอีก
1. เลขจำนวนเต็ม (-32767 ถึง +32767)	ไม่ใช้หน่วยความจำเพิ่มจากที่ใช้เป็นตำแหน่งตัวชี้เซลแล้ว
2. เลขจำนวนเต็ม (น้อยกว่า -32767 หรือมากกว่า +32767)	8 ไบต์
3. ค่าตัวเลขอื่น ๆ	8 ไบต์
4. ฟอร์แมตเซลที่ว่างเปล่า	ไม่เพิ่มหน่วยความจำ
5. ตัวอักษร	ต่ำสุด 6 ไบต์ และแปรค่าตามจำนวนตัวอักษร (ไบต์) รวมอักษรนำของตำแหน่งชิดซ้าย ชิดขวาหรืออยู่กลางกับอีก 1 ไบต์ ถ้าเป็นการเก็บในหน่วยความจำปกติ จะมีการปัดตัวเลขจำนวนไบต์รวมเป็นเลขคู่ และถ้าเก็บในหน่วยความจำเพิ่มขยายจะมีการปัดตัวเลขจำนวนไบต์เป็นจำนวนผลคูณของแปด
6. ชื่อกลุ่มเซล	34 ไบต์ โดยไม่ขึ้นกับจำนวนความยาวของชื่อ
7. ชื่อกราฟ	486 ไบต์ โดยไม่ขึ้นกับการเลือกัแบบของกราฟ
8. สูตรคำนวณ	ต่ำสุด 22 ไบต์ และแปรค่าตามความซับซ้อนของสูตรคำนวณ ถ้าเก็บในหน่วยความจำเพิ่มขยาย โปรแกรม 1-2-3 จะมีการปัดตัวเลขจำนวนไบต์เป็นจำนวนผลคูณของแปด
9. สูตรเชื่อมโยงเวอร์กชีต	การเชื่อมโยงไปยังไฟล์เวอร์กชีตครั้งแรก จะใช้ 25 ไบต์ รวมกับความยาวของชื่อไฟล์ ซึ่งจะมีการปัดตัวเลขให้เป็นเลขคู่ และในการเรียกเชื่อมโยงไปยังไฟล์เดิมครั้งต่อ ๆ ไปจะใช้อีก 32 ไบต์ทุกครั้งไป

ตัวอย่างการป้อนข้อมูลลงในเวอร์กชีต เพื่อศึกษาการใช้หน่วยความจำจะแสดงดังรูปที่ 3 โดยในตอนต้นหลังการเรียกโปรแกรม 1-2-3 จะเหลือพื้นที่หน่วยความจำอยู่ 331,568 ไบต์ เมื่อป้อนข้อมูลลงสู่เซล พื้นที่หน่วยความจำจะลดลงไปเรื่อย ๆ ซึ่งเราสามารถขอดูพื้นที่หน่วยความจำด้วยการเรียกใช้คำสั่ง /Worksheet Status ตามที่กล่าวมาแล้ว

การติดตั้งใช้งานโหมด WYSIWYG

หากเราต้องการให้ใช้งานในโหมด WYSIWYG เราจะต้องติดตั้งโปรแกรม Add-In ด้วยคำสั่ง /Add-In Attach แล้วเลือกไฟล์ WYSIWYG.ADN การติดตั้งใช้งานโหมด WYSIWYG นี้จะสิ้นเปลืองหน่วยความจำเป็นจำนวนมาก ซึ่งทั้งนี้จะขึ้นกับการเลือกใช้งานของเราเองด้วยว่า มีการใช้ฟอนต์จำนวนมากน้อยเพียงไร และเลือกขนาดฟอนต์ใหญ่หรือเล็กอย่างไร ในการทำงานของโหมด WYSIWYG ถ้าเราเลือกฟอนต์หลาย ๆ แบบ ฟอนต์แต่ละแบบจะต้องโหลดมาบรรจุในหน่วยความจำทำให้ใช้หน่วยความจำมากขึ้น โดยฟอนต์ขนาดโตตัวใหญ่ ๆ หรือมีพอยต์ (point) มาก ยังใช้หน่วยความจำมากกว่าฟอนต์ขนาดตัวเล็ก ๆ

ในโหมด WYSIWYG การเลือกฟอร์แมตเซลเป็นฟอนต์ต่างชนิดกันหรือต่างขนาดกัน จะมีการใช้พื้นที่หน่วยความจำปกติ 1 กิโลไบต์ และหน่วยความจำเพื่อโหลดฟอนต์นั้น ๆ มาลงอีกอย่างน้อย 2 กิโลไบต์ (สำหรับฟอนต์ขนาดเล็กที่สุด) ฟอนต์ขนาดใหญ่ขึ้นก็จะใช้หน่วยความจำสำหรับโหลดฟอนต์มากขึ้น ฟอนต์ขนาดใหญ่ที่สุดจะใช้พื้นที่หน่วยความจำถึง 14 กิโลไบต์ การฟอร์แมตเซลอื่น ๆ โดยใช้ฟอนต์เดิมขนาดเดิมที่เคยเรียกใช้แล้ว จะไม่สิ้นเปลืองหน่วยความจำเพิ่มอีก

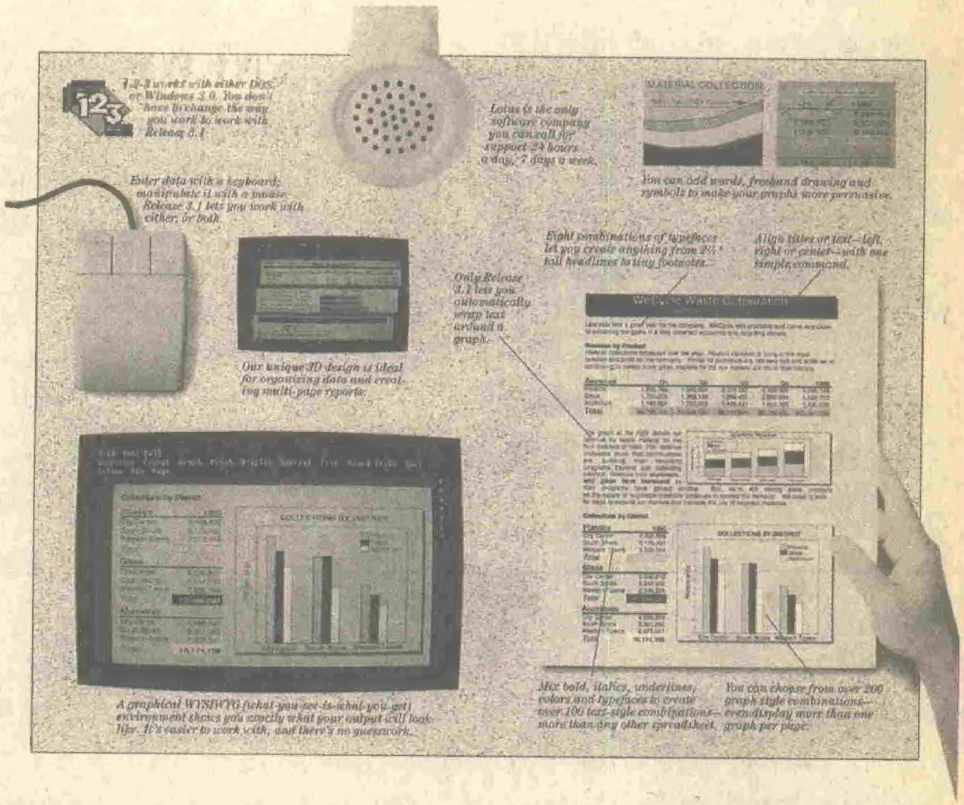
ในการเลือกแอตทริบิวต์ให้กับเซล ไม่ว่าจะเป็นตัวหนา ตัวเอียง ขีดเส้นใต้ การระบายแรเงา หรือการเลือกสี ก็จะเปลืองหน่วยความจำด้วยเช่นกัน โดยแอตทริบิวต์แรกจะใช้ 3 กิโลไบต์ และแอตทริบิวต์อื่น ๆ ต่อไปจะไม่ใช้หน่วยความจำเพิ่มอีก

การติดตั้งทำงาน UNDO

ตั้งแต่รีลีส 2.2 เป็นต้นไป โปรแกรม 1-2-3 จะเพิ่มลักษณะการทำงาน UNDO สำหรับเรียกชิ้นงานเดิมที่ได้สั่งให้ไว้แล้วกลับคืนมา เช่น เมื่อทำการลบกลุ่มเซล A1..A20 ไปแล้วโดยไม่ตั้งใจ และต้องการเรียกข้อมูลในกลุ่ม

เซลนั้นกลับคืนมา เราอาจเรียกข้อมูลนั้นคืนมาถ้ามีการใช้งาน UNDO ด้วย โดยโปรแกรม 1-2-3 จะต้องใช้พื้นที่หน่วยความจำมาเก็บสำเนาเวอร์กชีตตัวเดิมไว้ด้วย พื้นที่ที่ใช้สำรองจะเรียกว่า บัฟเฟอร์ UNDO

การติดตั้งลักษณะให้ทำงาน UNDO ได้ โลตัส 1-2-3 จะต้องสำรองเนื้อที่หน่วยความจำที่เหลืออยู่มากกว่าครึ่งหนึ่งเป็นบัฟเฟอร์ UNDO ซึ่งเมื่อหน่วยความจำถูกกันไว้ก่อนแล้ว พื้นที่หน่วยความจำจะเหลือน้อยลง จนเก็บข้อมูลเวอร์กชีตได้อีกไม่มากหรือใช้ติดตั้งโปรแกรม Add-In ต่าง ๆ ไม่ได้ กรณีถ้าในระบบมีหน่วยความจำเพิ่มขยายด้วย 1-2-3 จะสำรองบัฟเฟอร์ UNDO ในหน่วยความจำเพิ่มขยายขนาดเท่ากับหน่วยความจำปกติที่เหลือรวมกับหน่วยความจำเพิ่มขยายที่ใช้เก็บข้อมูลและตำแหน่งตัวชี้เซล รวมกับพื้นที่อีก 16 กิโลไบต์ เพื่อการสำรองข้อมูลภายในของ 1-2-3 ขนาดของบัฟเฟอร์ UNDO ในหน่วยความจำเพิ่มขยาย จะมีการปรับให้มีพื้นที่เป็นผลคูณของ 16 กิโลไบต์เสมอ



การจัดโครงสร้างเพื่อใช้หน่วยความจำน้อยสุด

การใช้พื้นที่หน่วยความจำของเวอร์กชีต เราสามารถควบคู่ให้น้อยลงมาก ๆ ด้วยการจัดตำแหน่งของเซลข้อมูล โดยพยายามให้ข้อมูลอยู่ติดกันในแนวคอลัมน์ ชุดของข้อมูลควรกระจายในแนวคอลัมน์ ไม่กระจายในแนวแถว เพราะจะเกิดแอคตีฟบล็อกมากกว่ากัน และเมื่ออยู่ในแนวคอลัมน์ เซลของข้อมูลควรจะจัดให้แอคตีฟบล็อกน้อยที่สุด และการจัดให้อยู่ใกล้ชิดกัน ไม่ควรเว้นช่องว่างระหว่างเซลข้อมูล ตัวอย่างเช่น การป้อนข้อมูลเลขจำนวนเต็มที่เซล A1 และป้อนข้อมูลอีกค่าที่เซล A8192 เพียงสองเซลเท่านั้น จะใช้พื้นที่หน่วยความจำถึง 32 กิโลไบต์ เพราะมีแอคตีฟบล็อกถึง 2048 บล็อก (8192 หารด้วย 4) บล็อกละ 16 ไบต์

ขณะใช้งานสร้างเวอร์กชีต หากมีตัวชี้บอกว่าหน่วยความจำกำลังจะไม่พอด้วยคำว่า MEM กะพริบ ๆ ตลอดที่บรรทัดล่างสุด เราอาจเพิ่มที่ว่างของหน่วยความจำมากขึ้น ด้วยการปฏิบัติดังนี้

- ลบข้อมูลที่ไม่จำเป็นออก ด้วยคำสั่ง /Range Erase ลบชื่อกลุ่มเซลด้วยคำสั่ง /Range Name Delete และลบชื่อกราฟด้วยคำสั่ง /Graph Name Delete ที่ไม่ใช้ออกเสียก่อน
- ลบคอลัมน์และแถวที่ไม่ใช้ออก
- ลดจำนวนฟอนต์ที่จะใช้ในเวอร์กชีต
- ตรวจสอบด้วยมีการฟอร์แมตเซลที่ว่างเปล่าไว้หรือเปล่า เพราะถึงแม้ว่าการฟอร์แมตเซลไม่ได้เพิ่มพื้นที่หน่วยความจำก็จริง แต่เป็นการทำให้บล็อกของเซลนั้นแอคตีฟ ซึ่งก็ยังเปลืองหน่วยความจำอยู่ ดังนั้นการฟอร์แมตกลุ่มเซลด้วย /Range Format ควรจะกำหนดเฉพาะกลุ่มเซลที่ต้องการใช้งานจริง ๆ
- เปลี่ยนสูตรคำนวณที่ให้ค่าเรียบร้อยแล้ว ไปเป็นค่าด้วยคำสั่ง /Range Value โดยค่าจากสูตรของเซลนั้นไม่มีการเปลี่ยนค่าอีก เพราะการใช้หน่วยความจำของสูตรคำนวณมากกว่าการใช้หน่วยความจำของข้อมูลที่เป็นค่า
- ยกเลิกการมีลักษณะการทำงานของ UNDO ที่ต้องสำรองหน่วยความจำเป็นบัฟเฟอร์ UNDO ด้วยคำสั่ง /Worksheet Global Default Other Undo Disable
- หากมีการใช้งานในโหมด WY-

SIWYG การเปลี่ยนการแสดงที่ไม่ใช่โหมดการพิมพ์จะช่วยประหยัดหน่วยความจำ ซึ่งเราอาจปรับเลือกใช้คำสั่ง : Display Options Adapter Auto ให้มีการเลือกใช้โหมดกราฟิกได้ถ้ามีหน่วยความจำเพียงพอสำหรับโหลดไดรเวอร์กราฟิกของ 1-2-3

- ปลดโปรแกรม Add-In บางโปรแกรมออกด้วยคำสั่ง /Add-In Detach

- ปลดโปรแกรมประเภทฝังในหน่วยความจำออก

- ตรวจสอบว่ามีการเรียกใช้โปรแกรม 1-2-3 ซ้อนหรือไม่ เพราะหากมีการออกจาก 1-2-3 มาที่ดอสชั่วคราวด้วยคำสั่ง /System แล้วจะกลับเข้า 1-2-3 ใหม่ เราจะพิมพ์คำสั่ง Exit แต่บางครั้งเราอาจเผลอพิมพ์เรียก 123 เข้าไปใหม่ ทำให้เรียกใช้โปรแกรมอีกครั้งหนึ่ง

- การเรียกโปรแกรม 1-2-3 ผ่านทางคำสั่ง Lotus ที่ตำแหน่งพร้อมรับคำสั่งของดอสจะเป็นการเรียกไปสู่โปรแกรมหลายตัว เช่น PrintGraph Translate 123 ซึ่งการใช้ 1-2-3 ด้วยการเลือกคำสั่งย่อย 123 จากเมนูนี้จะเปลืองหน่วยความจำอีก 2.5 กิโลไบต์ เราอาจหลีกเลี่ยงด้วยการเรียกใช้ 1-2-3 ด้วยการพิมพ์ 123 ที่ตำแหน่งพร้อมรับคำสั่งดอสได้ทันที

รูปที่ 3

ตัวอย่างที่ 1

1. หลังจากโหลดโปรแกรม 1-2-3 พื้นที่หน่วยความจำเหลืออีก 331,568 ไบต์

2. เริ่มป้อนค่าเลขจำนวนเต็ม 400 เข้าที่เซล A1 ตรวจสอบพื้นที่หน่วยความจำจะเหลือ 331,552 ไบต์ (แสดงว่ามีแอคตีฟบล็อก A1..A4 เปลืองพื้นที่ไป 16 ไบต์ สำหรับตำแหน่งตัวชี้เซล การป้อนค่าเลขจำนวนเต็ม 400 ไม่ได้เพิ่มหรือเปลืองหน่วยความจำอีก เพราะเลขจำนวนเต็มนั้นอยู่ระหว่าง -32767 ถึง +32767)

3. การเปลี่ยนแปลงแก้ไขค่าในเซล A1 เป็น +32768 หรือ A1 เป็น -32770 จะเหลือพื้นที่ 331,544 ไบต์ (แสดงว่าค่าตัวเลขนั้นอยู่นอกกรอบค่าระหว่าง +32767 และ +32767 ทำให้เปลืองเนื้อที่ 24 ไบต์ ที่มาจากพื้นที่แอคตีฟบล็อก 16 ไบต์ รวมกับจำนวนพื้นที่ที่ใช้มากขึ้นอีก 8 ไบต์ของเลขจำนวนเต็มนอกกรอบของข้อ 2)

4. การเปลี่ยนแปลงแก้ไขค่าในเซล A1 เป็นเลขมีจุดทศนิยม จะเหลือพื้นที่ 331,544 ไบต์ (แสดงว่าใช้พื้นที่ไป 24 ไบต์เท่ากับข้อ 3 โดยเป็นพื้นที่จะแอคตีฟบล็อก 16 ไบต์รวมกับพื้นที่ตัวเลขมีจุดทศนิยมจำเป็นต้องใช้เพิ่มอีก 8 ไบต์)

5. การลบค่าที่เซล A1 ด้วยการใช้คำสั่ง /Range Erase หรือการใช้ปุ่ม DEL จะยังคงเหลือพื้นที่ 331,552 ไบต์เท่าตอนแรก (แสดงว่าบล็อก A1..A4 ยังแอคตีฟอยู่ คือใช้พื้นที่ตำแหน่งตัวชี้เซลอยู่ 16 ไบต์เหมือนเดิม การลบค่าในเซลเป็นการเรียกใช้ฟอร์แมตที่ไม่ใช้พื้นที่หน่วยความจำเพิ่ม)

ตัวอย่างที่ 2

1. เริ่มต้นใหม่ด้วยการล้างเวอร์กชีต ด้วยคำสั่ง /Worksheet Erase กลับไปสู่ตำแหน่งที่เหลือพื้นที่หน่วยความจำ 331,568 ไบต์

2. การป้อนตัวอักษรเข้าที่เซล A1 ไม่ว่าจะเป็นตัวอักษรตัวเดียวคือ a ไปจนถึงตัวอักษร 4 ตัว ที่อาจเป็น abcd เมื่อรวมตัวอักษรนำและอีกหนึ่งไบต์ จะต้องเปลืองหน่วยความจำเมื่อตรวจสอบจะเหลือพื้นที่ 331,546 ไบต์ เป็นการใช้หน่วยความจำ 22 ไบต์ (แสดงว่า บล็อก A1..A4 แอคตีฟอยู่ใช้ 16 ไบต์ รวมกับการใช้หน่วยความจำเพิ่มของการป้อนตัวอักษรต่ำสุดอีก 6 ไบต์)

3. การเปลี่ยนแก้ไขตัวอักษรในเซล A1 เป็น abcde รวม 5 ตัวอักษร จะเหลือพื้นที่หน่วยความจำ 331, 544 ไบต์ เป็นการใช้หน่วยความจำเพิ่ม 24 ไบต์ (แสดงว่าบล็อก A1..A4 แอคตีฟอยู่เป็นการใช้หน่วยความจำ 16 ไบต์ รวมกับจำนวนไบต์ของตัวอักษร 5 ตัวรวมกับ 1 ไบต์ของตัวอักษรนำและ 1 ไบต์เสริมพิเศษ ซึ่งจริง ๆ แล้วหน่วยความจำของการป้อนตัวอักษรจะเป็น 7 ไบต์ แต่จะมีการปัดตัวเลขเป็นจำนวนคู่คือ 8 ไบต์เสียก่อน

4. การเปลี่ยนแก้ไขข้อมูลตัวอักษรในเซล A1 เป็นตัวอักษร 12 ตัว จะเหลือพื้นที่หน่วยความจำ 331,538 ไบต์เป็นการใช้พื้นที่ไป 30 ไบต์ (แสดงว่า บล็อก A1..A4 แอคตีฟอยู่ ใช้หน่วยความจำ 16 ไบต์ รวมกับ 12 ไบต์ของตัวอักษร 1 ไบต์ของตัวอักษรนำและ 1 ไบต์ของการเสริมพิเศษ เป็นจำนวนทั้งหมด 30 ไบต์พอดี)

เมื่อทำการลบข้อมูลที่ไม่จำเป็น ลบชื่อกลุ่มเซล ชื่อกราฟ ลบแถวและคอลัมน์ที่ไม่ใช้ออกแล้ว เราต้องทำการเซฟและเรียกเวอร์กชีตกลับมาใหม่เสมอ การลบแถวคอลัมน์หรือข้อมูลในเซลจะมีผลทำให้พื้นที่หน่วยความจำที่ปลดให้ว่างกระจายทั่วไปหมด ไม่ต่อเนื่องเป็นกลุ่มใหญ่เพื่อใช้ประโยชน์อื่น ๆ มากขึ้น การเซฟไฟล์เวอร์กชีตแล้วเรียกคืนมาโหลดใหม่ จะทำให้หน่วยความจำเรียงติดกัน ไม่ขาดเป็นช่วง ๆ ซึ่งจะทำให้มีโอกาสเรียกโปรแกรม Add-In มาใช้มากขึ้น และสร้างเวอร์กชีตขนาดใหญ่ขึ้น

สรุป

การเรียนรู้หรือเข้าใจการใช้หน่วยความจำของโลตัส 1-2-3 รีลีส 2.3 จะช่วยให้เราไม่มีปัญหาในเรื่องของการสร้างเวอร์กชีตใหญ่ ๆ และเรียกใช้โปรแกรมเสริมประเภท Add-In มากยิ่งขึ้น เพราะการป้อนข้อมูลตัวอักษรหรือตัวเลข หรือสูตรการคำนวณ ลงสู่เซลต่าง ๆ ของเวอร์กชีต จะใช้พื้นที่หน่วยความจำมากน้อยต่างกัน การจัดวางตำแหน่งของข้อมูลลงสู่เซลก็มีผลต่อการใช้หน่วยความจำของโลตัส 1-2-3 ด้วยเช่นกัน รายละเอียดต่าง ๆ เหล่านี้เชื่อว่าจะเป็นประโยชน์ต่อผู้ใช้ซอฟต์แวร์ตัวนี้ให้มีประสิทธิภาพมากยิ่งขึ้น

รูปที่ 3 (ต่อ)

ตัวอย่างที่ 3

1. ล้างเวอร์กชีตด้วยการใช้คำสั่ง /Worksheet Erase ทำให้มีพื้นที่หน่วยความจำเหลือ 331,568 ไบต์

2. ป้อนค่าเลขจำนวนเต็ม 400 ในเซล A1 และ 800 ในเซล A2 เหลือพื้นที่ 331,552 ไบต์ (มีการใส่ค่าในบล็อก A1..A4 แอคตีฟใช้พื้นที่หน่วยความจำ 16 ไบต์)

3. ทำการตั้งชื่อเซล A1 ให้ชื่อว่า test1 ด้วยคำสั่ง /Range Name Create ตรวจสอบพื้นที่หน่วยความจำจะเหลือเป็น 331,518 ไบต์ (แสดงว่า ใช้หน่วยความจำรวม 50 ไบต์ ของแอคตีฟบล็อก 16 ไบต์ และชื่อกลุ่มเซลอีก 34 ไบต์)

4. ตั้งชื่อเซล A2 ให้ชื่อว่า test2 จะเหลือพื้นที่ 331,484 ไบต์ (แสดงว่า ใช้หน่วยความจำ 84 ไบต์ ของแอคตีฟบล็อก 16 ไบต์กับชื่อของกลุ่มเซลสองชุดอีก 68 ไบต์)

5. ใช้คำสั่ง /Range Erase เซล A1 และ A2 ออก จะยังเหลือพื้นที่ 331,484 ไบต์เหมือนเดิม (แสดงว่า ชื่อของกลุ่มเซล test1 และ test2 ยังคงอยู่ในหน่วยความจำ)

6. ใช้คำสั่ง /Range Name Reset ลบชื่อกลุ่มเซลออกทั้งหมด ตรวจสอบหน่วยความจำจะเหลือพื้นที่ 331,552 ไบต์ (แสดงว่า บล็อก A1..A4 ยังแอคตีฟอยู่ จึงใช้พื้นที่หน่วยความจำ 16 ไบต์อยู่อีก)

ตัวอย่างที่ 4

1. ล้างเวอร์กชีตด้วยคำสั่ง /Worksheet Erase ทำให้มีที่ว่าง 331,568 ไบต์เหมือนเดิม

2. ป้อนข้อมูลในสองคอลัมน์ เพื่อใช้สร้างกราฟโดยเซลต่าง ๆ มีค่าดังนี้

B5 : JAN	C5 : 12
B6 : FEB	C6 : 34
B7 : MAR	C7 : 56
B8 : APR	C8 : 78

ตรวจสอบพื้นที่เหลือ 331,512 ไบต์

(มาจากแอคตีฟบล็อกสองบล็อกใช้ 32 ไบต์ รวมกับค่าต่ำสุด 6 ไบต์ของเซลตัวอักษร 4 เซล อีก 24 ไบต์ รวมเป็น 56 ไบต์)

3. ใช้คำสั่ง /Graph กำหนดชนิดของกราฟเป็นกราฟแท่ง ข้อมูลในแกน X เป็นเซล B5..B8 และแกน Y ของข้อมูล A เป็นเซล C5..C8 ซึ่งเมื่อ View หรือขอดูกราฟจะได้เป็นกราฟแท่งตามต้องการ ตรวจสอบหน่วยความจำยังคงเหลือ 331,512 ไบต์เช่นเดิม

4. ใช้คำสั่ง /Graph Name Create ตั้งชื่อกราฟแท่งเป็น BAR ตรวจสอบหน่วยความจำ จะเหลือ 330,044 ไบต์ (แสดงว่าการตั้งชื่อกราฟ จะใช้พื้นที่ 468 ไบต์)

5. ใช้คำสั่ง /Graph Name Reset เพื่อลบชื่อกราฟออก หน่วยความจำจะเหลือ 331,512 ไบต์ เหมือนข้อ 3 ตอนต้นนั่นเอง